***Ejercicio 1***

***MER:***

Adjuntamos en la carpeta el archivo .drawio y su visualización en un archivo .html

**Proceso de construcción del MER:**

En primer lugar, nos centramos en identificar las entidades principales y las relaciones, como por ejemplo usuario asistente virtual. En primera instancia nos pareció una buena idea hacer que el asistente virtual sea una entidad débil de usuario, pero esto nos causó algunas dudas como por ejemplo si el existente virtual dependía expresamente del usuario podría traer complicaciones como la transferencia de asistentes virtuales o lo reutilización de los mismos. O por ejemplo si un usuario elimina su cuenta, nos parecía que no era adecuado borrar con el su asistente virtual, por lo que decidimos hacerlo una entidad independiente. Después nos fuimos centrando en ir haciendo los diferentes subsistemas que indica la letra por separado para posteriormente unificarlos todos.

Una de las decisiones que nos costó mucho fue la “gestión de interacciones”. En una primera instancia se nos ocurrió que podríamos hacer una agregación en la relación de usuario-asistente virtual y esa agregación relacionarla con una entidad que sea interacción que contenga los datos de la interacción, pero nos encontramos con el problema de que la relación de usuario-asistente no es una relación de N a N pues un usuario solo tiene un asistente virtual entonces hacer una agregación acá no nos pareció viable. La solución que encontramos fue crear una relación “interacciona” entre el usuario y el asistente y a esta relación atribuirle los datos que especificaba la letra. De esta forma se puede llevar una trazabilidad organizada de las interacciones.

Otro punto que nos llevo tiempo de resolver fue el tema de que los usuarios que tienen la suscripción gratuita solo pueden acceder al rol de “amigo”. Llegamos a la conclusión que no había forma de representarlo en el modelo, así que decidimos representarlo mediante una restricción no estructural.

Una corrección que hicimos es que habíamos hecho que la billetera fuera una entidad débil del usuario, sin embargo, analizándolo más en profundidad logramos llegar a la conclusión que debería ser una entidad fuerte ya que el numero de las billeteras debería ser único.   
Para el caso de la obtención del certificado de logro, decidimos simplemente crear una entidad que sea certificado de logro, crear una relación “recibe” que relaciona al usuario con el certificado. Habíamos pensado crear una agregación con el usuario el idioma y relacionarlo con la obtención del certificado, pero nos pareció irrelevante ya que la una relación que consideramos significativa fue que el usuario reciba su certificado.

Luego para relacionar los subsistemas de la billetera y la configuración del avatar habíamos pensado en hacer una agregación con billetera y usuario pero nuevamente nos pareció innecesario y decidimos relacionar la billetera directamente con los artículos que puede comprar así como apariencia, voz, rasgos, intereses, etc. Nos pareció innecesario ya que la entidad billetera ya “conoce” al usuario.

RESTRICCIONES NO ESTRUCTURALES:

1. En usuario, los valores que pueden tomar los rangos de edad son: “18-24”, “25-34”,

“35-44”, “45-54”, “55-64”, “65 o más”

2. El número de teléfono del usuario tiene que ser único

3. El Género del asistente virtual puede tomar los valores: femenino, masculino o no binario

4. El tipo de voz que tiene la entidad "Voz" puede ser: "femenina" o "masculina"

5. El tono de la voz puede tomar los valores: alegre, calmo, seguro, energético u optimista

6. Los nombres que toma el rasgo de personalidad pueden ser: (seguro, tímido, energético, práctico, soñador, artístico o lógico)

7. La categoría de la apariencia puede tomar los valores: (cabello, piel, cuerpo, ojos, maquillaje y los accesorios como piercing y tatuajes)

8. El perfil que puede tomar el avatar tiene las opciones "gratuito" y "de pago"

9. Si el usuario tiene una suscripción gratuita, el asistente solo podrá tener el rol "Amigo"

10. El tipo de interacción que puede tener un usuario con el asistente toma los valores: (consulta, tutoría, conversación

casual, coaching)

11. El nivel de dominio que pueden tomar los certificados de logro son: (‘Básico’, ‘Intermedio’ o ‘Avanzado’).

12. Si el usuario adquiere la suscrpcion gratuita el método de pago es nulo.

***Ejercicio 2***

Empleado(NroEmp, Nombre, FechaNacimiento, FechaIngreso, Calle, NroPuerta, Apartamento, )

PK:NroEmp

Administrativo(NroOficina, NroEmpleado, TipoAdministrativo)

PK: NroOficina

FK: NroEmpleado ->Empleado

Mantenimiento(NroEmp, TipoTrabajo)

PK: NroEmp

FK: NroEmp -> Empleado

Docente(NroEmp)

PK: NroEmp

FK: NroEmp -> Empleado

Biblioteca(Direccion)

PK: Direccion

Libro(Codigo, Nombre)

PK: Codigo

Libro\_Autores(Codigo, Autor)

PK: Codigo, Autor

FK: Codigo -> Libro

Alumno(TipoDocumento, NroDocumento, NroAlumno, TipoAlumno, NombreCompleto, Edad)

PK: NroAlumno

Alumno\_Telefono(NroAlumno, Telefono)

PK: NroAlumno, Telefono

FK: NroAlumno -> Alumno

Intercambio(Numero, UniversidadOrigen, UniversidadDestino)

PK: Numero

Salón(NroSalon, Capacidad)

PK:NroSalon

Curso(Codigo, Descripcion)

PK:Codigo

Dictado(Codigo, NumDic, Semestre, FechaInicio)

PK: Codigo, NumDic

FK: Codigo->Curso

dicta(NumEmp, NumDic, Codigo)

PK: NumEmp, NumDic, Codigo

FK: NumEmp->Empleado/Docente

FK: NumDic->Dictado

FK: Codigo->curso

rol\_dicta(NumEmp, NumDic, Codigo, Rol)

PK: NumEmp, NumDic, Codigo, Rol

FK: NumEmp, NumDic, Codigo->trabaja

asigna(NroSalon, NroEmp, NumDic, Codigo)

PK:NroSalon, NroEmp, NumDic, Codigo

FK: NroEmp, NumDic, Codigo->dicta

FK: NroSalon -> Salon

posee(Direccion, Codigo)

PK: Direccion, Codigo

FK: Direccion -> Biblioteca

FK: Codigo ->Libro

utiliza(CodigoCurso, CodigoLibro)

PK: CodigoCurso, CodigoLibro

FK: CodigoCurso -> Curso

FK:CodigoLibro -> Libro

realiza(NroAlumno, Numero, Desde, Hasta)

PK: NroAlumno

FK: NroAlumno -> Alumno

FK: Numero -> Intercambio

reserva(NroAlumno, Codigo)

PK: NroAlumno, Codigo

FK: NroAlumno -> Alumno

FK: Codigo -> Libro

fechaReserva\_reserva(NroAlumno, Codigo, fechaReserva)

PK: NroAlumno, Codigo, fechaReserva  
FK: NroAlumno, Codigo -> reserva

presta(NroAlumno, Codigo)

PK: NroAlumno, Codigo

FK: NroAlumno -> Alumno

FK: Codigo -> Libro

fecha\_presta(NroAlumno, Codigo, Prestamo, Devolucion)

PK: NroAlumno, Codigo, Prestamo, Devolucion

FK: NroAlumno, Codigo -> presta

Supervisa(supervisor, supervisado)

PK: supervisor

FK: supervisor -> empleado

FK: supervisado -> empleado

**Restricciones no estructurales:**1. La dirección es un campo obligatorio para cada empleado. Si vive en un apartamento, el número de apartamento es opcional.

2. El tipo de para empleado de mantenimiento debe tomar uno de los siguientes valores: “pintura”, “electricidad” o “jardinería”.

3. El valor del atributo “extranjero” para docentes solo puede tomar los valores “Si” o “No”.

4. La edad de los estudiantes debe ser mayor a 17.

5. Los tipos de documento valido para los alumnos son: “cedula de identidad”, “pasaporte” o “DNI”.

6. El atributo tipoAlumno toma valores “local” o “extranjero”.

7. Si un alumno local realiza un intercambio el valor del atributo UniversidadOrigen quedara vacío.

8. Los alumnos de intercambio solo pueden cursar dictados durante el periodo especificado del intercambio.

9. El resultado de los dictados debe estar entre los valores 0 y 100.

10. El valor del atributo capacidad de los salones debe estar comprendido entre 20 y 40.

11. El valor del código de los libros debe ser un alfanumérico comprendido por tres letras y cuatro números.

12. Todos los campos fecha deben tener el formato de: DD-MM-YYYY